



# Prueba técnica

Operations

*Clasificación de privacidad: Documento para uso interno*

# Prueba de Programación Para Desarrollador Java/SQL

## Introducción

Este documento presenta el ejercicio de programación a realizar por los candidatos para los puestos de desarrolladores Java y SQL en Minderest.

La idea es plantear un sencillo ejercicio de programación con el fin de verificar las aptitudes de desarrollo, nivel de atención a detalles, calidad del código etc. de los candidatos.

No se buscan “genialidades” ni soluciones particularmente ingeniosas, no hay trampa. Esta prueba es exclusivamente una referencia más como parte de proceso de selección y su finalidad es principalmente confirmar que los candidatos tienen un nivel tecnológico mínimo.

**IMPORTANTE:** El lenguaje de trabajo en Minderest será Java , pero este ejercicio se puede ejecutar tanto en Java como en otros lenguajes similares, como por ejemplo C# o incluso C++ .

A continuación se describe el dominio de la prueba y cuál es la funcionalidad a implementar.

### **Modelo de Dominio**

Imaginemos una empresa que gestiona información de Productos de venta online. A su vez, esta empresa tiene Clientes que venden productos y les interesa conocer cuáles de sus productos son también vendidos por otros clientes .

Un Cliente tiene como mínimo:

- Un nombre arbitrario.
- Un identificador textual único que no puede repetirse en el sistema para uso interno.

Un Producto de Cliente tiene como mínimo:

- Un nombre que es único por cliente.

Entonces: los productos tienen un atributo principal “nombre”. Puede darse la circunstancia de que distintos clientes vendan el mismo producto con nombres diferentes. Por ejemplo, Worten puede vender “Monitor DELL UltraSharp 24” y MediaMarkt “Pantalla UltraSharp Dell 24”, tratándose realmente del mismo producto.

De ahí que se quiera tener reflejada una relación entre productos de forma que podamos vincular unos productos con otros, significando “estos productos son el mismo”.

Las condiciones son:

- Un producto no puede estar relacionado con (“ser el mismo que”) otro del mismo cliente. A un cliente sólo le interesa conocer los productos que “son el mismo” en otros clientes. Es decir: no se permite que productos de un mismo cliente estén relacionados entre sí como iguales.
- Un producto de un cliente puede “ser el mismo” que uno o más de otro(s) cliente(s).

Se Pretende

Usando las siguientes tecnologías:

- Java/C#/C++/...
- MySQL 5.7 o superior (o cualquier otro motor de BBDD relacional SQL)

Programar lo siguiente:

- Base de Datos:
  - Modelo de tablas, relaciones, etc. (el esquema de la BDD) de acuerdo a lo explicado arriba como requisitos mínimos.
  - Script (o programa en Java, C#, ...) para pre- “rellenar” 10 clientes.
  - El “output” debería ser un archivo SQL con todo lo necesario.
- Una clase de “Servicio” que implemente métodos con la siguiente funcionalidad:
  - Insertar un nuevo producto. Idealmente, que contemple los requisitos expuestos también en el apartado del Modelo de Dominio.
  - A partir de un cliente y un producto, devolver una lista con los posibles “mismos productos” en otros clientes.
  - Para un producto de un cliente y otro producto que se ha decidido “son el mismo”, establecer en la BDD esta relación de equivalencia o “igualdad”.